Utilizing the Normalized Laplacian Matrix to Map their Respective Eigenvalues to Sound Frequencies

1.1 Introduction

Spectral graph theory is the study of the properties of the Laplacian matrix, adjacently matrix, eigenvectors and eigenvalues associated with a graph. The way these properties interact with one another tells us a lot about the graph itself. One particular property of these graphs is that the range of the eigenvalues and eigenvectors are mapped between 1 and 2. This unique property is what we will focus on this paper. We will map the eigenvalues to frequencies. We see how the properties of the normalized Laplacian matrix affect the sound produced.

1.2 Julia Language

We used Julia to perform the data analysis and write the code to create the frequencies. Julia was developed in 2012 by Jeff Bezanson, Stefan Karpinski, Viral B. Shah, and Alan Edelman. The language was developed to be both high-level and fast. Julia is very useful for low level systems programing in addition to being a general-purpose programming language.

Julia was especially advantageous for this project due its matrix manipulation libraries that made it possible to compute the eigenvalues. While python possess similar libraries, the computation time compared to Julia.

A study compared Julia to python and found out that Julia ran times faster than python. The following is a general comparison between both languages.

Julia vs. python

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Language | Speed |  |  |  |  |  |
| Julia |  |  |  |  |  |  |
| Python |  |  |  |  |  |  |
| C++ |  |  |  |  |  |  |
| Javascript |  |  |  |  |  |  |

While Julia has many advantages it is still a new language and undeveloped. It still requires more libraries to be able to compete with languages like python. Currently, Julia has \_\_\_\_\_ libraries compared to python \_\_\_\_ libraries. \_\_\_\_ is currently working on creating more libraries that will aid in data analysis, programing and math modeling.

1.3 Code Development

In order to create the frequencies, the eigen values of the normalized Laplacian matrix need to be performed.

The function normalized Laplacian matrix takes in graph g and returns the Sparse arrays of an \_\_\_\_. It does this by passing graph G through the Combinatorial adjacency. The graph is further passed through the.

Once we have the normalized Laplacian matrix, the eigenvalues can be calculated. In the function normalized Laplacian spectrum, the matrixed is passed through the eigenvals attribute from the library LinAlg.

The obtained eigenvalues are mapped to frequencies. Based on the eigenvalues of matrix we map them from 0 to 2 with each. A 0 eigenvalue gets mapped to a zero frequency, 1 gets mapped to ‘zero frequencies’, 2 gets mapped to zero frequency plus 2 octaves.

We finalize the process by converting the frequencies to tones. With the function make tones, the spectrum and frequencies are used to covert\_\_\_ .

Developing the sound is the first part of the research. Understanding how each sound was constructed is the second part. The properties of the graph themselves and they influence the sound.

1.4 Data analysis

Each graph produced a distinct sound given their graph properties. The graphs were labeled based on a description of the sound. We used the terms length and pattern to label the sound. For the duration of the sound, a sound was either labeled short, medium, or long length. For pattern of the sound, a rising tone was labeled rising, otherwise it was labeled complex. At the end the final labels were short rising, short complex, medium rising, medium complex, long rising, and long complex.

We wanted to find out if each type of sound had unique properties that could be replicated to produce graphs that could produce sound like previous graphs. The different graphs were filtered and grouped by their sound and pattern. One of the first observations that we made was that most of the graphs were medium rising. Out of the 84 graphs, 34% of the graphs sampled were medium rising. A medium rising graph had as average the following: \_\_\_\_, \_\_\_, \_\_\_, \_\_\_, and \_\_\_.

We next looked at each individual graph based on their sound. The graphs were filtered and categorized based on their sound description. Once the graphs, were separated, it was easier to analyze each type of graph individually.

The scatter plot showed correlation between some the properties. For instance, in medium rising graphs, the higher/lower the domination number was the higher/lower the would be. The following chart shows the findings of all the graphs.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type | Domination number | Radius |  |  |  |
|  | Higher | Lower | No correlation | Higher | Higher |
|  |  |  |  |  |  |

We further broke down the graphs based on two criteria: Length and patterns. By looking at each individual factor, we could determine if there were some properties that influenced the sound.

Looking at the length, the graphs were sorted and categorized by their length: short, medium, and long. Out of 81 graphs,61 are long length, \_\_\_ are medium length, and \_\_\_\_ are short length.. The following chart describes the behavior of the properties as their values would increase/decrease.

Out of the 8 of short length, one is rising and seven are complex sound. Out of the 11 of medium length, 8 are rising and 3 are complex. And out of the 61 of long length, 38 are rising and 23 are complex.

Our findings revealed that length was mostly determined by \_\_\_\_, \_\_\_\_\_, \_\_\_\_, while patterns were determined by \_\_\_\_\_\_\_. Combined they produced a distinct sound that reflected their respective characteristics.

Machine learning

Questions answered:

what is the typical domination number of each type of graph?

What happens to the \_\_\_ as the domination increases/decreases?

What is the average domination of the graphs?

What properties do sounds with medium complex produce?

what sound is produced more frequently?

What are the properties of sounds based on their patterns?

What are the properties of sounds that based on their length?

What specific property affects the length of the sound?

What specific property affects the pattern of the sound?

If such property exist, what happens to other properties as such property increases/decreases?

Independent set – set S of vertices such that for every two vertices in S, there is no edge connecting the two.

Domination number – is the subset that every vertex not in D is adjacent to at least one member of D.

Zero forcing number, of a graph is the minimum cardinality of a set S of black vertices wheras vertices in V(G) such that V(G) is turned black after finitely many applications of the

Diameter – it is the greatest distance between any pair of vertices.

Radius – the minimum among all the maximum distances between a vertex to all other vertices.

Order – number of vertices in the graph

Chromatic number – mimimal number of colours needed to colour the vertices in such a way that no two adjacent have the same colour.

Clique – subset of vertices of an undirected graph such that every two distint vertices in the clique are adjacent.

Annihilation number of a(G) is the largest interger k such that sum of the first k terms of the non-decreasing degree sequence of G is at most the number of its edges.

What are properties of the different types of grahps

Conclusion

Graphs with high independent number produce short complex sounds. Theses graph with high independent values have small eigenvalues.